UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/691,888	10/23/2003	Timothy P. McKee	MFCP.110115	8996

45809   7590   10/17/2007

SHOOK, HARDY & BACON L.L.P.
(c/o MICROSOFT CORPORATION)
INTELLECTUAL PROPERTY DEPARTMENT
2555 GRAND BOULEVARD
KANSAS CITY, MO 64108-2613

EXAMINER
LE, MIRANDA

ART UNIT	PAPER NUMBER
2167	

MAIL DATE	DELIVERY MODE
10/17/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

PTOL-90A (Rev. 04/07)

	Application No.	Applicant(s)
Office Action Summary	10/691,888	TIMOTHY MCKEE
	Examiner	**Art Unit**	
	Miranda Le	2167	

-- *The MAILING DATE of this communication appears on the cover sheet with the correspondence address* --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _02 August 2007_.

2a)☒ This action is **FINAL**.   2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-28_ is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _1-28_ is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All   b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☐ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO/SB/08)
    Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____.

5)☐ Notice of Informal Patent Application

6)☐ Other: _____.

U.S. Patent and Trademark Office

PTOL-326 (Rev. 08-06)   Office Action Summary   Part of Paper No./Mail Date 20071010

DETAILED ACTION

1.   This communication is responsive to Amendment, filed 08/02/2007.

Claims 1-28 are pending in this application. This action is made Final.

2.·   The objection to the specification (drawings, claim objection) of the invention has been

withdrawn in view of the amendment.

Claim Rejections - 35 USC § 103

3.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in
> section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are
> such that the subject matter as a whole would have been obvious at the time the invention was made to a person
> having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the
> manner in which the invention was made.

This application currently names joint inventors. In considering patentability of the
claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various
claims was commonly owned at the time any inventions covered therein were made absent any
evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out
the inventor and invention dates of each claim that was not commonly owned at the time a later
invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c)
and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

4.   Claims 1-28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Sullivan

(U.S. Patent No. 5,737,557), in view of Cosic (U.S. Patent No. 7,117,225).

As per claim 1, Sullivan teaches a computer system for presenting related items in a data

storage device to a user, the system comprising:

a data storage device *(i.e. storage media 20, col. 5, lines 1-21)* containing a plurality of

items *(i.e. for multiple application programs, documents and/or data files (hereinafter*

collectively designated "items"), (col. 3, lines 14-26) stored in accordance with a data schema

and containing relational information corresponding to at least a portion of said plurality of items

(i.e. a storage element in which is stored contextual information for defining the appearance and

behavior of the software suite window and the icons displayed therein and for identifying the

locations of files associated with the icons, col. 3, lines 27-35), wherein the relational

information allows relationships between two or more the plurality of items to be determined

(i.e. each software suite has associated therewith a storage element in which is stored contextual

information for defining the appearance and behavior of the software suite window and the icons

displayed therein and for identifying the locations of files associated with the icons, col. 3, lines

27-35), wherein said data storage device utilizes said relational information to delete one or more

items from the data storage device in response to a change in at least a portion of said

relationships *(i.e. The user could similarly choose to deinstall the application and remove the*

icon from the display. In this instance all files recorded in the storage element that are

associated with the icon would be deleted and the icon itself would be deleted, col. 11, lines 3-

33); and

a shell *(i.e. graphical display element, col. 5, lines 1-21)* for presenting said plurality of

items to a user wherein the shell is configured to present a selected item to a user *(i.e. the user*

can select or specify one or more individual icons in order to effect some action on the individual

icon(s) and the item(s) it represents., col. 7, lines 22-32) and is further configured to utilize said

relational information to present one or more items *(i.e. As used herein, the term "software suite"*

refers to a compound computer display object that provides a single integrated visual

representation of multiple application programs or files, hereinafter collectively referred to as

"items." FIG. 1A illustrates one method by which a software suite of the present invention may

be accessed, col. 5, lines 1-21) in said data storage device which are related to said selected item *(i.e. Yet another technical advantage achieved with the invention is that it enables a user to conceptualize and manage a collection of items as a single title, thereby simplifying the user's mental model of the system, col. 4, lines 26-29).*

Sullivan does not specifically teach universal data.

Cosic teaches a universal data storage device *(i.e. a universal data management interface (UDMI) system. This aspect includes a processing system that executes instructions to generate a visual interface through which a user can access, manage, and manipulate data on plural different types of remote databases, col. 2, lines 40-45).*

It would have been obvious to one of ordinary skill of the art having the teaching of Sullivan and Cosic at the time the invention was made to modify the system of Sullivan to include the limitations as taught by Cosic.

One of ordinary skill in the art would be motivated to make this combination in order to allow multiple users to access, manage, and manipulate data within each of the multiple standard database management systems in view of Cosic *(col.2, lines 51-56)*, as doing so would give the added benefit of manipulating data on plural different types of remote databases as taught by Cosic *(col. 2, lines 40-45).*

As per claim 9, Sullivan teaches a computer-implemented method for presenting related items in a data storage device *(i.e. storage media 20, col. 5, lines 1-21)* to a user, the method comprising:

accessing data in said data storage device *(i.e. storage media 20, col. 5, lines 1-21)*,

wherein said data storage device stores a plurality of items *(i.e. for multiple application*

programs, documents and/or data files (hereinafter collectively designated "items"), (col. 3, lines

14-26) in accordance with a data schema, and wherein at least a portion of said plurality of items

contain relational information which allows relationships between said plurality of items to be

determined *(i.e. As used herein, the term "software suite" refers to a compound computer display*

object that provides a single integrated visual representation of multiple application programs

or files, hereinafter collectively referred to as "items." FIG. 1A illustrates one method by which a

software suite of the present invention may be accessed, col. 5, lines 1-21);

utilizing said relational information to determine a relationship between a selected item

(i.e. the user can select or specify one or more individual icons in order to effect some action on

the individual icon(s) and the item(s) it represents, col. 7, lines 22-32) and one or more of the

items containing said relational information in the data storage device *(i.e. Yet another technical*

advantage achieved with the invention is that it enables a user to conceptualize and manage a

collection of items as a single title, thereby simplifying the user's mental model of the system, col.

4, lines 26-29);

displaying said selected item and one or more related items to the user *(i.e. the user can*

select or specify one or more individual icons in order to effect some action on the individual

icon(s) and the item(s) it represents., col. 7, lines 22-32, Figs. 1A, 1B);

receiving a user input causing a change *(i.e. The user could similarly choose to deinstall*

the application and remove the icon from the display, col. 11, lines 3-33) in said relationship *(i.e.*

the user may copy or move icons between software suites or to a location external to the system

of the present invention, for example, a desktop, while maintaining with the icon(s) all contextual

information stored relative to that icon, col. 3, lines 53-67); and

deleting at least one of said plurality of items from said data storage device in response to

said change *(i.e. The user could similarly choose to deinstall the application and remove the icon*

from the display. In this instance all files recorded in the storage element that are associated

with the icon would be deleted and the icon itself would be deleted, col. 11, lines 3-33).

Sullivan does not specifically teach universal data.

Cosic teaches a universal data storage device *(i.e. a universal data management interface*

(UDMI) system. This aspect includes a processing system that executes instructions to generate a

visual interface through which a user can access, manage, and manipulate data on plural

different types of remote databases, col. 2, lines 40-45).

It would have been obvious to one of ordinary skill of the art having the teaching of

Sullivan and Cosic at the time the invention was made to modify the system of Sullivan to

include the limitations as taught by Cosic.

One of ordinary skill in the art would be motivated to make this combination in order to

allow multiple users to access, manage, and manipulate data within each of the multiple standard

database management systems in view of Cosic *(col.2, lines 51-56),* as doing so would give the

added benefit of manipulating data on plural different types of remote databases as taught by

Cosic *(col. 2, lines 40-45).*

As per claim 13, Sullivan teaches one or more computer-readable media having computer-executable instructions for performing a method for presenting related items in a data storage device *(i.e. storage media 20, col. 5, lines 1-21)* to a user, the method comprising:

accessing data in said data storage device *(i.e. storage media 20, col. 5, lines 1-21)*, wherein said data storage device stores a plurality of items in accordance with a data scheme *(i.e. a storage element in which is stored contextual information for defining the appearance and behavior of the software suite window and the icons displayed therein and for identifying the locations of files associated with the icons, col. 3, lines 27-35)*, and wherein at least a portion of said plurality of items contain relational information which allows relationships between two or more of said plurality of items to be determined, wherein at least a portion of said relationships designate one or more source items and one or more target items *(i.e. each software suite has associated therewith a storage element in which is stored contextual information for defining the appearance and behavior of the software suite window and the icons displayed therein and for identifying the locations of files associated with the icons, col. 3, lines 27-35)*;

utilizing said relational information to determine a relationship between a selected item · *(i.e. the user can select or specify one or more individual icons in order to effect some action on the individual icon(s) and the item(s) it represents., col. 7, lines 22-32)* and one or more of the items containing said relational information in the data storage device *(i.e. Yet another technical advantage achieved with the invention is that it enables a user to conceptualize and manage a collection of items as a single title, thereby simplifying the user's mental model of the system, col. 4, lines 26-29)*;

presenting said selected item and one or more related items to the user *(i.e. the user can select or specify one or more individual icons in order to effect some action on the individual icon(s) and the item(s) it represents., col. 7, lines 22-32, Figs. 1A, 1B);*

receiving a user input *(i.e. user input and further enables user input to be applied to the graphic representation the group of items, i.e., the icon(s) and or software suite without requiring knowledge on the part of the user of individual files and storage locations and without requiring the use of external deinstallation/reinstallation utility programs, col. 6, lines 56 to col. 7, line 2)* altering at least one of said one or more source items or altering at least a portion of said relationships *(i.e. the system enables a user, with minimal input, to initiate certain operations simultaneously upon all of the items represented in the software suite col. 3, lines 36-52);* and

deleting at least one of said one or more target items from said data store if said at least one target item is not related to at least one of said one or more source items *(i.e. The user could similarly choose to deinstall the application and remove the icon from the display. In this instance all files recorded in the storage element that are associated with the icon would be deleted and the icon itself would be deleted, col. 11, lines 3-33).*

Sullivan does not fairly teach universal data.

Cosic teaches a universal data storage device *(i.e. a universal data management interface (UDMI) system. This aspect includes a processing system that executes instructions to generate a visual interface through which a user can access, manage, and manipulate data on plural different types of remote databases, col. 2, lines 40-45).*

It would have been obvious to one of ordinary skill of the art having the teaching of

Sullivan and Cosic at the time the invention was made to modify the system of Sullivan to

include the limitations as taught by Cosic.

One of ordinary skill in the art would be motivated to make this combination in order to

allow multiple users to access, manage, and manipulate data within each of the multiple standard

database management systems in view of Cosic *(col.2, lines 51-56)*, as doing so would give the

added benefit of manipulating data on plural different types of remote databases as taught by

Cosic *(col. 2, lines 40-45)*.

As per claim 18, Sullivan teaches a shell for presenting related items in a data storage

device to a user, the shell comprising:

a data storage device *(i.e. storage media 20, col. 5, lines 1-21)* interaction component

which retrieves data associated with one or more items from the data storage device *(i.e. a*

storage element in which is stored contextual information for defining the appearance and

behavior of the software suite window and the icons displayed therein and for identifying the

locations of files associated with the icons, col. 3, lines 27-35), wherein said one or more items

are stored in accordance with a data schema and at least a portion of said one or more items

contain relational information that allows relationships between two or more items to be

determined *(i.e. each software suite has associated therewith a storage element in which is*

stored contextual information for defining the appearance and behavior of the software suite

window and the icons displayed therein and for identifying the locations of files associated with

the icons, col. 3, lines 27-35);

a related item presentation component which utilizes said retrieved data to present related

items to a user *(i.e. the system enables a user, with minimal input, to initiate certain operations*

simultaneously upon all of the items represented in the software suite col. 3, lines 36-52),

wherein the relationship presentation component is configured to present a selected item to a user

(i.e. the user can select or specify one or more individual icons in order to effect some action on

the individual icon(s) and the item(s) it represents., col. 7, lines 22-32, Figs. 1A, 1B) and is

further configured to utilize said relational information to present one or more items in said data

storage device which are related to said selected item *(i.e. each software suite has associated*

therewith a storage element in which is stored contextual information for defining the

appearance and behavior of the software suite window and the icons displayed therein and for

identifying the locations of files associated with the icons, col. 3, lines 27-35).

Sullivan does not teach:

universal data;

wherein at least a portion of said relationships has associated life-time management

semantics time;

an item life-time management control which utilizes said associate life-time management

semantics to delete one or more items from the universal data storage device in response to a

change in at least a portion of said relationships.

Cosic teaches:

a universal data storage device *(i.e. a universal data management interface (UDMI)*

system. This aspect includes a processing system that executes instructions to generate a visual

interface through which a user can access, manage, and manipulate data on plural different

types of remote databases, col. 2, lines 40-45)

wherein at least a portion of said relationships has associated life-time management

semantics time *(i.e. if the inactivity time is less than the allowed inactivity time (e.g., 15 minutes),*

the UDMI Engine Program updates the corresponding user's session record in the LastVisited

column with the new "last" time (expressed in seconds since 1970) the user executed an UDMI

function, col. 13, lines 63-67);

an item life-time management control which utilizes said associate life-time management

semantics to delete one or more items from the universal data storage device in response to a

change in at least a portion of said relationships *(i.e. if the inactivity time is longer than the*

allowed inactivity time, the UDMI Engine Program 1000 deletes the user's session record, and

presents a "you have been inactive for too long" message through GUI. The user then has an

option to go to the initial login screen, col. 14, lines 1-5).

It would have been obvious to one of ordinary skill of the art having the teaching of

Sullivan and Cosic at the time the invention was made to modify the system of Sullivan to

include the limitations as taught by Cosic.

One of ordinary skill in the art would be motivated to make this combination in order to

delete the user's session record in view of Cosic *(col. 14, lines 1-5),* as doing so would give the

added benefit of manipulating data on plural different types of remote databases as taught by

Cosic *(col. 2, lines 40-45).*

As per claim 25, Sullivan teaches a computer system for presenting related items in a data storage device *(i.e. storage media 20, col. 5, lines 1-21)* to a user, the method comprising:

means for accessing data in said data storage device *(i.e. storage media 20, col. 5, lines 1-21)*, wherein said data storage device stores a plurality of items *(i.e. for multiple application programs, documents and/or data files (hereinafter collectively designated "items"), (col. 3, lines 14-26)* in accordance with a data scheme, and wherein at least a portion of said plurality of items contain relational information which allows relationships between said plurality of items to be determined *(i.e. As used herein, the term "software suite" refers to a compound computer display object that provides a single integrated visual representation of multiple application programs or files, hereinafter collectively referred to as "items." FIG. 1A illustrates one method by which a software suite of the present invention may be accessed, col. 5, lines 1-21)*;

means for utilizing said relational information to determine a relationship between a selected item *(i.e. the user can select or specify one or more individual icons in order to effect some action on the individual icon(s) and the item(s) it represents., col. 7, lines 22-32)* and one or more of the items containing said relational information in the data storage device *(i.e. Yet another technical advantage achieved with the invention is that it enables a user to conceptualize and manage a collection of items as a single title, thereby simplifying the user's mental model of the system, col. 4, lines 26-29)*;

means for displaying said selected item and one or more related items to the user *(i.e. the user can select or specify one or more individual icons in order to effect some action on the individual icon(s) and the item(s) it represents., col. 7, lines 22-32, Figs. 1A, 1B)*;

means for receiving a user input causing a change *(i.e. The user could similarly choose to deinstall the application and remove the icon from the display, col. 11, lines 3-33)* in said relationship *(i.e. the user may copy or move icons between software suites or to a location external to the system of the present invention, for example, a desktop, while maintaining with the icon(s) all contextual information stored relative to that icon, col. 3, lines 53-67)*; and

means for deleting at least one of said plurality of items from said data storage device in response to said change *(i.e. The user could similarly choose to deinstall the application and remove the icon from the display. In this instance all files recorded in the storage element that are associated with the icon would be deleted and the icon itself would be deleted, col. 11, lines 3-33)*.

Sullivan does not specifically teach universal data.

Cosic teaches a universal data storage device *(i.e. a universal data management interface (UDMI) system. This aspect includes a processing system that executes instructions to generate a visual interface through which a user can access, manage, and manipulate data on plural different types of remote databases, col. 2, lines 40-45)*.

It would have been obvious to one of ordinary skill of the art having the teaching of Sullivan and Cosic at the time the invention was made to modify the system of Sullivan to include the limitations as taught by Cosic.

One of ordinary skill in the art would be motivated to make this combination in order to allow multiple users to access, manage, and manipulate data within each of the multiple standard database management systems in view of Cosic *(col.2, lines 51-56)*, as doing so would give the

added benefit of manipulating data on plural different types of remote databases as taught by

Cosic *(col. 2, lines 40-45).*

As to claims 2, 19, 26, Sullivan teaches the relational information corresponding to one

or more of said plurality of items includes a set of item characteristics *(i.e. the name of and*

general information concerning the software product, and a list of all files installed when the

software product was installed, regardless of their location, and can deinstall or reinstall the

entire software product with minimal effort, col. 3, lines 36-52).

As to claims 3, 27, Sullivan teaches said shell is configured to present one or more of

said set of item characteristics to a user *(i.e. Such actions include displaying the directory in*

which the software product has been installed; displaying the name of the software product and

general product information; displaying a list of all files installed when the software product was

installed, regardless of their location; and deinstalling and reinstalling a software product, col.

7, lines 12-21).

As per claim 4, Sullivan teaches said shell is configured to accept a user input

representing a selection to view one or more items in the data storage device having one of said

item characteristics *(i.e. the user can select or specify one or more individual icons in order to*

effect some action on the individual icon(s) and the item(s) it represents. These actions include,

for example deleting the icon and the item it represents, moving or copying the icon and the item

it represents to another location, placing the icon and the item it represents on the GUI

clipboard, running the executable program assigned to the icon; and displaying special

information or any special status of the item, such as setup program, multimedia content, main

executable, help file, etc., col. 7, lines 22-32).

As per claim 5, Sullivan teaches said shell is configured to present one or more items in

the data storage device which share one of said item characteristics *(i.e. a compound computer*

display object that provides a single integrated visual representation, via a "software suite

window," for multiple application programs, documents and/or data files (hereinafter

collectively designated "items"), which are represented in the suite window by icons. In addition,

a software suite window may contain iconized representations of other software suites (col. 3,

lines 14-26).

As per claim 6, Sullivan teaches the shell is configured to present at least a portion of

said relational information *(i.e. a compound computer display object that provides a single*

integrated visual representation, via a "software suite window," for multiple application

programs, documents and/or data files (hereinafter collectively designated "items"), which are

represented in the suite window by icons. In addition, a software suite window may contain

iconized representations of other software suites (col. 3, lines 14-26).

As per claim 7, Sullivan teaches the shell is configured to accept a user input

representing a selection to view items in the data storage device which are related to said selected

item *(i.e. the user can select or specify one or more individual icons in order to effect some*

action on the individual icon(s) and the item(s) it represents. These actions include, for example deleting the icon and the item it represents, moving or copying the icon and the item it represents to another location, placing the icon and the item it represents on the GUI clipboard, running the executable program assigned to the icon; and displaying special information or any special status of the item, such as setup program, multimedia content, main executable, help file, etc., col. 7, lines 22-32).

As per claim 8, Sullivan teaches said relational information corresponding to the selected item includes a set of item characteristics associated with the selected item and wherein said user input represents a selection to view one or more items in the data storage device which share one of said set of item characteristics with the selected item *(i.e. the user can select or specify one or more individual icons in order to effect some action on the individual icon(s) and the item(s) it represents. These actions include, for example deleting the icon and the item it represents, moving or copying the icon and the item it represents to another location, placing the icon and the item it represents on the GUI clipboard, running the executable program assigned to the icon; and displaying special information or any special status of the item, such as setup program, multimedia content, main executable, help file, etc., col. 7, lines 22-32).*

As to claims 10, Sullivan teaches the displaying of said selected item and one or more related items includes displaying at least a portion of said relational information to a user *(i.e. the user can select or specify one or more individual icons in order to effect some action on the individual icon(s) and the item(s) it represents. These actions include, for example deleting the*

icon and the item it represents, moving or copying the icon and the item it represents to another

location, placing the icon and the item it represents on the GUI clipboard, running the

executable program assigned to the icon; and displaying special information or any special

status of the item, such as setup program, multimedia content, main executable, help file, etc.,

col. 7, lines 22-32).

As to claims 11, Sullivan teaches said method further comprises receiving a user input

representing a selection to view one or more items in the data storage device which are related to

said selected item *(i.e. the user can select or specify one or more individual icons in order to*

effect some action on the individual icon(s) and the item(s) it represents. These actions include,

for example deleting the icon and the item it represents, moving or copying the icon and the item

it represents to another location, placing the icon and the item it represents on the GUI

clipboard, running the executable program assigned to the icon; and displaying special

information or any special status of the item, such as setup program, multimedia content, main

executable, help file, etc., col. 7, lines 22-32).

As per claim 12, Sullivan teaches the displaying of said selected item and one or more

related items is responsive to said input *(i.e. the user can select or specify one or more individual*

icons in order to effect some action on the individual icon(s) and the item(s) it represents. These

actions include, for example deleting the icon and the item it represents, moving or copying the

icon and the item it represents to another location, placing the icon and the item it represents on

the GUI clipboard, running the executable program assigned to the icon; and displaying special

information or any special status of the item, such as setup program, multimedia content, main

executable, help file, etc., col. 7, lines 22-32).

As per claim 14, Sullivan teaches the relational information includes a set of item

characteristics *(i.e. the name of and general information concerning the software product, and a*

list of all files installed when the software product was installed, regardless of their location, and

can deinstall or reinstall the entire software product with minimal effort, col. 3, lines 36-52).

As per claim 15, Sullivan teaches accessing data in said data storage device is in

response to a user input representing a selection to view one or more items in the data storage

device which are related to said selected item *(i.e. the user can select or specify one or more*

individual icons in order to effect some action on the individual icon(s) and the item(s) it

represents. These actions include, for example deleting the icon and the item it represents,

moving or copying the icon and the item it represents to another location, placing the icon and

the item it represents on the GUI clipboard, running the executable program assigned to the

icon; and displaying special information or any special status of the item, such as setup

program, multimedia content, main executable, help file, etc., col. 7, lines 22-32).

Cosic teaches a universal data storage device *(i.e. a universal data management interface*

(UDMI) system. This aspect includes a processing system that executes instructions to generate a

visual interface through which a user can access, manage, and manipulate data on plural

different types of remote databases, col. 2, lines 40-45).

As per claim 16, Sullivan teaches said relational information corresponding to the

selected item includes a set of item characteristics associated with the selected item and wherein

said user input represents a selection to view one or more items in the data storage device which

share one or more item characteristics with the selected item *(i.e. the user can select or specify*

one or more individual icons in order to effect some action on the individual icon(s) and the

item(s) it represents. These actions include, for example deleting the icon and the item it

represents, moving or copying the icon and the item it represents to another location, placing the

icon and the item it represents on the GUI clipboard, running the executable program assigned

to the icon; and displaying special information or any special status of the item, such as setup

program, multimedia content, main executable, help file, etc., col. 7, lines 22-32).

As per claim 17, Sullivan teaches the presenting of said selected item and one or more

related items includes presenting at least a portion of said relational information to a user *(i.e. the*

user can select or specify one or more individual icons in order to effect some action on the

individual icon(s) and the item(s) it represents. These actions include, for example deleting the

icon and the item it represents, moving or copying the icon and the item it represents to another

location, placing the icon and the item it represents on the GUI clipboard, running the

executable program assigned to the icon; and displaying special information or any special

status of the item, such as setup program, multimedia content, main executable, help file, etc.,

col. 7, lines 22-32).

As per claim 20, Sullivan teaches said related item presentation component is configured to present one or more of said set of item characteristics to a user *(i.e. the user can select or specify one or more individual icons in order to effect some action on the individual icon(s) and the item(s) it represents. These actions include, for example deleting the icon and the item it represents, moving or copying the icon and the item it represents to another location, placing the icon and the item it represents on the GUI clipboard, running the executable program assigned to the icon; and displaying special information or any special status of the item, such as setup program, multimedia content, main executable, help file, etc., col. 7, lines 22-32).*

As per claim 21, Sullivan teaches said related item presentation component is configured to present one or more items in the data storage device which share one of said item characteristics *(i.e. the user can select or specify one or more individual icons in order to effect some action on the individual icon(s) and the item(s) it represents. These actions include, for example deleting the icon and the item it represents, moving or copying the icon and the item it represents to another location, placing the icon and the item it represents on the GUI clipboard, running the executable program assigned to the icon; and displaying special information or any special status of the item, such as setup program, multimedia content, main executable, help file, etc., col. 7, lines 22-32).*

As per claim 22, Sullivan teaches the related item presentation component is configured to present at least a portion of said relational information *(i.e. the user can select or specify one or more individual icons in order to effect some action on the individual icon(s) and the item(s) it*

represents. These actions include, for example deleting the icon and the item it represents, moving or copying the icon and the item it represents to another location, placing the icon and the item it represents on the GUI clipboard, running the executable program assigned to the icon; and displaying special information or any special status of the item, such as setup program, multimedia content, main executable, help file, etc., col. 7, lines 22-32).

As per claim 23, Sullivan teaches the shell is configured to accept a user input representing a selection to view items in the data storage device which are related to said selected item *(i.e. user input and further enables user input to be applied to the graphic representation the group of items, i.e., the icon(s) and or software suite without requiring knowledge on the part of the user of individual files and storage locations and without requiring the use of external deinstallation/reinstallation utility programs, col. 6, lines 56 to col. 7, line 2).*

As per claim 24, Sullivan teaches said relational information corresponding to the selected item includes a set of item characteristics associated with the selected item and wherein said user input *(i.e. user input and further enables user input to be applied to the graphic representation the group of items, i.e., the icon(s) and or software suite without requiring knowledge on the part of the user of individual files and storage locations and without requiring the use of external deinstallation/reinstallation utility programs, col. 6, lines 56 to col. 7, line 2)* represents a selection to view one or more items in the data storage device which share one of said set of item characteristics with the selected item *(i.e. the user can select or specify one or more individual icons in order to effect some action on the individual icon(s) and the item(s) it*

represents. These actions include, for example deleting the icon and the item it represents,

moving or copying the icon and the item it represents to another location, placing the icon and

the item it represents on the GUI clipboard, running the executable program assigned to the

icon; and displaying special information or any special status of the item, such as setup

program, multimedia content, main executable, help file, etc., col. 7, lines 22-32).

As per claim 28, Sullivan teaches said means for accessing data in said data storage

device interacts with said data storage device in response to a user input representing a selection

to view one or more items in the data storage device having one of said item characteristics *(i.e.*

the user can select or specify one or more individual icons in order to effect some action on the

individual icon(s) and the item(s) it represents. These actions include, for example deleting the

icon and the item it represents, moving or copying the icon and the item it represents to another

location, placing the icon and the item it represents on the GUI clipboard, running the

executable program assigned to the icon; and displaying special information or any special

status of the item, such as setup program, multimedia content, main executable, help file, etc.,

col. 7, lines 22-32).

Cosic teaches a universal data storage device *(i.e. a universal data management interface*

(UDMI) system. This aspect includes a processing system that executes instructions to generate a

visual interface through which a user can access, manage, and manipulate data on plural

different types of remote databases, col. 2, lines 40-45).

Response to Arguments

5.   Applicant's arguments filed 08/02/2007 have been fully considered but they are not

persuasive as for the following reasons:

Applicant argues that: **Sullivan discloses deleting files once a user chooses to uninstall**

an application, Sullivan is silent as to techniques for deleting files when the relationship

among the files have changed.

The examiner disagrees. Claim 9 recites:

(a) receiving a user input causing a change in said relationship; and

(b) deleting at least one of said plurality of items from said universal data storage device

in response to said change.

Sullivan reads on the claimed limitation as follows:

receiving a user .... equates to the step of *"the system enables a user to delete, move*

and/or copy some or all of the icons", See col. 3, lines 36-52.

a change in said relationship is interpreted as "deinstalled a file" *(i.e. the system*

facilitates the deinstallation of files represented by a particular suite window or icon, See col. 3,

lines 53-67). This step equates to the step (a) of the claim invention. Note that "the relationship"

corresponds to the association between the icon and files of Sullivan *(i.e. the locations of files*

associated with the icons, See col. 3, lines 27-35).

deleting at least one .... equates to the step of *"the window or icon and the file(s) it*

represents may be deleted altogether, in which case the represented file(s) will also be deleted",

See col. 3, lines 53-67; and, **at least one of said plurality of items** of the claim invention

equates to the window or icon and the files... of Sullivan.

In response to the Applicant's argument regarding claim rejection under 35 U.S.C 103(a),

as detailed, the claim language as presented is still read on by the Sullivan reference at the cited

paragraph in the claim rejections. Sullivan does teach all Applicant's claim limitation: 1. a

computer-implemented method for presenting related items in a **data storage device** to a user

(i.e. storage media 20, col. 5, lines 1-21). 2. accessing data in said data storage device *(i.e.*

storage media 20, col. 5, lines 1-21), wherein said data storage device stores **a plurality of**

items *(i.e. for multiple application programs, documents and/or data files (hereinafter*

collectively designated "items"), (col. 3, lines 14-26) in accordance with a data schema, and

wherein at least a portion of said **plurality of items contain relational information which**

allows relationships between said plurality of items to be determined *(i.e. the term "software*

suite" refers to a compound computer display object that provides a single integrated visual

representation of multiple application programs or files, hereinafter collectively referred to as

"items." FIG. 1A illustrates one method by which a software suite of the present invention may

be accessed, col. 5, lines 1-21);

   utilizing said relational information to determine a relationship between a selected

item *(i.e. the user can select or specify one or more individual icons in order to effect some*

action on the individual icon(s) and the item(s) it represents, col. 7, lines 22-32) and one or

more of the items containing said relational information in the data storage device *(i.e. a*

technique that enables a user to conceptualize and manage a collection of items as a single

title, thereby simplifying the user's mental model of the system, col. 4, lines 26-29);

   displaying said selected item and one or more related items to the user *(i.e. the user*

can select or specify one or more individual icons in order to effect some action on the

individual icon(s) and the item(s) it represents, col. 7, lines 22-32, Figs. 1A, 1B);

   receiving a user input causing a change *(i.e. The user could similarly choose to*

deinstall the application and remove the icon from the display, col. 11, lines 3-33) in said

relationship *(i.e. the user may copy or move icons between software suites or to a location*

external to the system of the present invention, for example, a desktop, while maintaining with

the icon(s) all contextual information stored relative to that icon, col. 3, lines 53-67); and

   deleting at least one of said plurality of items from said data storage device **in**

response to said change *(i.e. The user could similarly choose to deinstall the application and*

remove the icon from the display. In this instance all files recorded in the storage element that

are associated with the icon would be deleted and the icon itself would be deleted, col. 11, lines

3-33).

   3. Sullivan does not specifically teach universal data.

   Cosic teaches this limitation as *a universal data management interface (UDMI) system*

which includes a processing system that executes instructions to generate a visual interface

through which a user can access, manage, and manipulate data on plural different types of

remote databases, col. 2, lines 40-45). Thus, since both references are directed to methods and

systems of generating a graphical user interface and managing data via the graphical user

interface, it would have been obvious to one of ordinary skill of the art at the time of the invention

to combine Sullivan and Cosic in arriving at the instant invention because it would enhance the

graphical user interfaces ("GUIs") which provide an intuitive interface to assist users in utilizing

the system's functionality as taught by Sullivan, and further provides the ability to assist the user

in establishing a link to the database, obtaining the information via the link, synchronizing with

the data structure of the database, and/or managing and manipulating data in the database

following accessing, as taught by Cosic.

Therefore, a case for obviousness is established by combining the Sullivan reference with

Cosic as the combination teaches all the claim limitations.

Furthermore, applicant's arguments regarding the deletion of files in Sullivan is triggered

by a user input ordering an application to be uninstalled; or Sullivan selects the file for deletion

by relying on static association with the to-be-deleted application, have been considered but they

are irrelevant; it is noted that the claim language as presented neither specify "automatically

deleting" nor the step of deleting cannot be performed by a user; plus, there is nothing in the

claim language prohibits the relation based on "static association" or "static relationship".

Accordingly, the claimed invention as represented in the claims does not represent a

patentable over the art of record.

Conclusion

6.   **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time

policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE

MONTHS from the mailing date of this action. In the event a first reply is filed within TWO

MONTHS of the mailing date of this final action and the advisory action is not mailed until after

the end of the THREE-MONTH shortened statutory period, then the shortened statutory period

will expire on the date the advisory action is mailed, and any extension fee pursuant to 37

CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event,

however, will the statutory period for reply expire later than SIX MONTHS from the mailing

date of this final action.

　　　　Any inquiry concerning this communication or earlier communications from the examiner should be directed to Miranda Le whose telephone number is (571) 272-4112. The examiner can normally be reached on Monday through Friday from 8:30 AM to 5:00 PM.

　　　　If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John R. Cottingham, can be reached on (571) 272-7079. The fax number to this Art Unit is 571-273-8300.

　　　　Any inquiry of a general nature or relating to the status of this application should be directed to the Group receptionist whose telephone number is (703) 305-3900.

　　　　Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Miranda Le
October 10, 2007
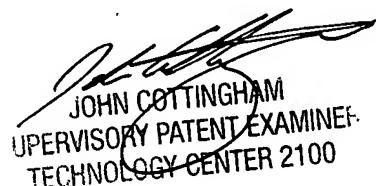
JOHN COTTINGHAM
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100